Chapter 5

Matrix relevance learning for multi-class classification with spectral data

Abstract

We discuss the use of matrix relevance learning, a popular extension to prototype learning algorithms, applied to a three-class classification task of diagnosing cassava diseases using spectral data. Previously this diagnosis has been done using plant image data taken with a smartphone. However, for this method, disease symptoms need to be visible and, once symptoms have manifested, the root of the plant is already affected and can no longer be used as food. This research is premised on the hypothesis that diseased crops without visible symptoms can be detected using spectral information, allowing for early action measures. We analyze visible and near-infrared spectra captured from leaves infected with two common cassava diseases (cassava brown streak disease and cassava mosaic virus disease) found in Sub-Saharan Africa as well as from healthy plants. The spectral data come with thousands of dimensions, therefore different wavelengths are analyzed in order to identify the most relevant spectral bands. To cope with the nominally high number of input dimensions of data, functional decomposition of the spectra is considered. The outlined classification task is addressed using Generalized Matrix Relevance Learning Vector Quantization and compared with the standard classification techniques performed in the space of expansion coefficients.
5.1 Introduction

The ability to quickly diagnose disease in the field is of critical importance in most agro-reliant economies the world over. For places where the crop is not only of economic but also food security importance, this is particularly crucial. In this chapter we further investigate improved ways of accurately diagnosing plants in the field by leveraging a unique dataset: spectral data from plant leaves and using improved algorithms that not only provide higher accuracy but also a profile of wavelengths that are most important for the classification.

We particularly focus on cassava (Manihot esculenta), a staple crop in Sub-Saharan Africa that feeds over 500 million people daily. Cassava suffers from two serious diseases: cassava brown streak disease (CBSD) and cassava mosaic virus disease (CMD). According to (Zeyimo et al. 2019), CBSD and CMD together account for over 90% of yield losses in cassava production systems in Sub-Saharan Africa. This in turn greatly affects smallholder farmers.

We present an approach to detection and diagnosis of CBSD and CMD based on image spectroscopy to extract representative features from example leaves manifesting these diseases, and machine learning for building the predictive models based on such data. This work is an early step in our endeavor to run experiments on diseased but non-symptomatic cassava plants using spectral data collected from spots on a diseased leaf that are symptomatic and spots that are non-symptomatic. The novelty in our approach is not only applying spectroscopy in field-level diagnostics of cassava but also optimizing the models for fewer features of the data while maintaining accuracy. This is very important for deployment. Our eventual goal is to deploy these models in low-cost sensor devices to capture spectral data from the leaves and low-cost smartphone devices to run the built models on these datasets.

The field of feature engineering and feature selection provides techniques for reducing the number of required features of input data, usually making the model simpler and less prone to bias from noise in unnecessary features. Here we employ Generalized Matrix Relevance Learning Vector Quantization (GMLVQ), a highly intuitive algorithm that can be optimized in the training to give high accuracy and also detect the most important features relevant for the classification.

We specifically apply GMLVQ here for a number of reasons: It has been used successfully in previous, related studies and displayed favorable performance (Schneider et al. 2008, Melchert et al. 2016a, Melchert et al. 2019). It is, however, not our aim to show that GMLVQ outperforms other classifiers in the problem at hand. Prototype-based systems in general are natural tools for the analysis of multi-class datasets. GMLVQ is particularly suitable for the combination with efficient dimensional reduction methods in terms of functional representations or PCA. As outlined in sec-
5.1. Introduction

tion 5.2, the actual GMVLQ training is implemented efficiently in the corresponding coefficient space without losing other favorable features of prototype-based learning. Most importantly, GMVLQ offers great interpretability and insight into the importance of features for the classification and can serve as a tool for the identification of relevant spectral wavelengths. Here, we exploit these important aspects to a large extent as outlined in greater detail in section 5.2.1.

The sections that follow describe the experimental procedure we applied to provide evidence of the efficacy of using spectral data for this task and optimizing the models for a reduced featureset of the data. First, we give a small synopsis of the literature related to the use of spectroscopy for classification and some examples, section 5.2 describes the GMVLQ algorithm, section 5.2.1 discusses feature selection and dimensionality reduction, section 5.3.1 describes the experimental set-up we employed and results and discussion follow in section 5.4 and 5.5 respectively.

A common way to diagnose crops in the field has been leave images, for instance taken with a smartphone. Several recent studies have demonstrated the efficacy of these methods on visual diagnosis of different crops. This work builds on previous studies in (Adowo et al. 2010, Tuhaise et al. 2014, Owomugisha and Mwebaze 2016) and (Mwebaze and Biehl 2016) that focused on the use of conventional smartphone camera plant images to diagnose disease in the field.

Image data are prone to several challenges including occlusion, varied lighting conditions, varied image background effects and effects of scale and depth that require rigorous training to get robust algorithms. Spectral data tend not to be affected by these factors and have found favor with several researchers. Examples of successful use of spectral data in early disease detection is found in some of the following work: Fluorescence spectroscopy to detect mechanical and disease stresses in citrus plants is used in (Belasque et al. 2008). A similar methodology was also employed for detection of diseases in citrus plants in the USA and Brazil (Wetterich et al. 2013). Methods for early detection of rice blast disease using near-infrared hyper-spectral imaging are presented (Yang et al. 2012). Some research has focused on a combination of pests and diseases, for example (Feng, Liao, Liang, Zhao and Dai 2009) employed multispectral imaging methodology for the diagnosis of plant diseases and insect pests. The use of near infrared spectroscopy to analyze cold rice blast was discussed (Tan et al. 2012). Hyper-spectral data were used for the pre-symptomatic detection of infections in sugar beet (Arens et al. 2016).

The uniqueness of our work is the extension toward the early detection of diseases in plants based on spectral data. Using images only works once the diseases have manifested physically on leaves of the diseased plant. Using hyper-spectral imaging techniques is infeasible in our context because of the costs associated with acquiring the hyper-spectral cameras. Spectrometry gives us the opportunity to de-
tect disease before it is symptomatic giving the small holder farmer in our case a window of time to apply an intervention to control the disease.

This work is an extension of previous chapters that investigated diagnosing cassava diseases using spectral data from visibly infected leaves compared with use of image-based features extracted from crop leaves taken by a mobile camera. A key aspect of this earlier work was to understand whether the location of where spectra was taken from a leaf matters, particularly if there is a significant difference between taking spectral data from visibly infected parts of the leaf or from parts of the leaf that are not visibly infected. Results of that study showed a significant increase in performance for the experiment using the spectral data from the good part.

Another aspect of spectral data is that it is very high-dimensional, nominally comprising for instance more than 3600 features or dimensions. A feature pre-processing step is thus essential in this case to ensure our models do not suffer from the large \( p \) small \( n \) problem.

To do this, we apply and compare different processing strategies and machine learning frameworks for the actual diagnosis task. As the baseline approach, we consider the use of the original high-dimensional data. To reduce data dimensionality, we employ the functional representation of spectra in terms of polynomial approximations and formulate the machine learning in the space of the corresponding coefficients. The basic approach is introduced and investigated using spectral data from different contexts and more general functional data (Melchert et al. 2016a, Melchert et al. 2016b).

We also employ standard principal component analysis (PCA) as a dimension reduction technique and compare it to the other schemes.

The ultimate aim is to identify a specific feature representation or particular features (i.e. wavelengths or ranges of wavelengths) which contain most information for the classification and that will facilitate technical solutions using simple sensors.

The selection of features is mainly addressed within the example framework of GMLVQ, e.g. (Schneider et al. 2007, Schneider et al. 2009a). This prototype and distance-based classifier was previously studied for detecting cassava diseases on the basis of relatively few features directly derived from camera images in (Mwebaze et al. 2011, Mwebaze et al. 2015, Mwebaze and Biehl 2016). We apply a similar methodology here to show the comparative advantage of the different featureset.
5.2 The GMLVQ machine learning framework

In chapter 2, we introduced a family of LVQ and in this area, we give a brief on GMLVQ algorithm that outperformed other classifiers and also used for feature selection.

Considering a dataset of the form:

$$\{x^\mu, y^\mu\}^p_{\mu=1}$$

(5.1)

where $x^\mu \in \mathbb{R}^N$ are feature vectors and the labels $y^\mu \in 1, 2, ...C$ specify their class membership.

These data are generally standardized by performing a z-score operation as shown in Eq. (5.2). This is computed by subtracting the sample mean ($\vartheta$) from the target data point ($x^\mu$) and dividing by the target standard deviation ($\delta$):

$$z_i = \frac{x_i^\mu - \vartheta_i}{\delta_i}$$

(5.2)

where $i \in \{1, 2, ..., N\}$

From chapter 2 we defined a general LVQ system as a set of $M$ prototype vectors $w^j \in \mathbb{R}^N$ which carry labels $c(w^j) \in \{1, 2, ..., C\}$ such that $W = \{w^j, c(w^j)\}_{j=1}^M$.

A nearest prototype classifier (NPC) assigns a given feature vector $x \in \mathbb{R}^N$ to the closest prototype with respect to some meaningful distance measure.

From equation 2.2, adaptive distance $d^A$ is used in GMLVQ thus distance measure $d^A(x, w)$ is defined as:

$$d^A(x, w) = (x - w)^T \Lambda (x - w)$$

(5.3)

As suggested in (Sato and Yamada 1995), GMVLQ training is guided by the optimization of a cost function:

$$E(W) = \sum_{\mu=1}^p \Phi \left( \frac{d^A_j(x_i) - d^K_j(x_i)}{d^A_j(x_i) + d^K_j(x_i)} \right).$$

(5.4)

In this work, we use the publicly available LVQ toolbox (Biehl 2017), which implements the batch gradient minimization of the cost function, Eq. (6.4), with adaptive step size control (Biehl 2017, Papari et al. 2011). If not specified otherwise, we use default parameters as suggested in (Biehl 2017). Training yields the GMLVQ classifier in terms of the prototype vectors and the relevance matrix $\Lambda$. Its diagonal elements $\Lambda_{ii}$ can be interpreted as the relevance of the corresponding feature dimensions for the classification (Schneider et al. 2007, Biehl et al. 2016).
5.2.1 Dimensionality reduction

Spectral data of the type considered here are nominally high-dimensional. As a consequence, the naive application of machine learning techniques will result in classifiers with a very large number of adjustable parameters, which causes problems ranging from computationally expensive training to a potentially increased risk of over-fitting.

The former is disadvantageous for efficient deployment of the model, for instance in mobile systems. The latter point could result in inferior generalization performance.

It is important to realize that spectra, like other functional data, comprise highly correlated features, as such the intensities of neighboring wavelengths can be expected to be very similar in a more or less smooth spectrum.

We consider three different approaches for the dimensionality reduction of the data in order to circumvent the above mentioned problems.

In approach one, we use functional approximation by Chebyshev polynomials and approach we apply PCA also used in Chapter 4.

A key aspect we investigated in this study was the optimal number of coefficients required to represent the spectrum. To do this, we performed back transformation of features from the coefficient space to the original spectral space.

Both dimension reduction schemes can be interpreted as a linear transformation of the general form:

\[ y = \Psi x \] where \( x \in \mathbb{R}^N \), \( \Psi \in \mathbb{R}^{M \times N} \) and \( y \in \mathbb{R}^M \) \hspace{1cm} (5.5)

which projects the original data, potentially centered in the case of PCA, to an \( M \)-dimensional space with \( M < N \).

A low-dimensional \( y \) corresponds to \( M \) expansion coefficients in the polynomial representation. When applying PCA, the components of \( y \) are the projections of \( x \) on the \( M \) leading principal components.

We can compare the form of the distance measure in both spaces:

\[ (v - y)^\top \hat{\Lambda} (v - y) = (w - x)^\top \Psi^\top \hat{\Lambda} \Psi (w - x) \equiv (w - x)^\top \Lambda (w - x). \] \hspace{1cm} (5.6)

Here we denote a prototype and the relevance matrix in the low-dimensional space by \( v \in \mathbb{R}^M \) and \( \hat{\Lambda} \in \mathbb{R}^{M \times M} \), respectively. We observe that formally

\[ v = \Psi w \] and \( \hat{\Lambda} = \Psi^\top \hat{\Lambda} \Psi. \] \hspace{1cm} (5.7)
Hence, we can back-transform the relevance matrix $\hat{\Lambda}$ and although the training is performed in terms of Chebyshev coefficients or principal components, we can identify the relevance of the original features, i.e. in terms of wavelengths or ranges thereof.

**Peak selection**

Peak selection was our third approach to dimensionality reduction. The method selects a set of wavelengths with the highest peaks from the relevance profile obtained from running GMLVQ on the original data spectrum. It has a very simple intuition; the wavelengths with higher peaks represent areas in the spectrum where the sensor had a strong response to the item being measured. Harvesting wavelengths where there is a high response for the different classes provides an intuitive way of selecting features that may be relevant. This technique has commonly been used in many signal-processing applications, e.g. (Liutkus 2015). Like the two methods (PCA and Chebyshev polynomials) mentioned above, a subset of the original dimensions is selected by constructing new dimensions.

Given our feature matrix $\Lambda$, we put an intensity threshold on $\Lambda_{ii}$ to eliminate low-ranked features and select out wavelengths with a response above the threshold. We employ the convenient function `findpeaks` defined in MATLAB(R2016a) which works by finding local peaks or valleys (local extrema) in a noisy vector using a user-defined magnitude threshold to determine if each peak is significantly larger (or smaller) than the data around it.

### 5.3 Experiments

Here we discuss the experiments carried out with spectra collected from leaves of cassava plants and how the methods described were applied. First we describe collection of the spectral data and the pre-processing applied. Next we present the machine learning techniques and discuss how they are combined with polynomial expansions and PCA.

#### 5.3.1 Experiment design and data collection

Our goal was to collect representative spectral data from the leaves of cassava plants under two conditions: when plants are healthy and when they are infected by the two different diseases CBSD and CMD with visibly symptomatic leaves. In Fig. 5.2, we provide some example images of leaves from the two diseases. The manifestation of disease on the leaf is determined to large extent by the variety of cassava and
the severity of disease. In future work one goal is to relate the spectra extracted to the severity of disease. For this current work however, we only looked at the binary case, disease vs (visibly)healthy for the two diseases.

These data were acquired using a CI-710 miniature leaf spectrometer (CID Bio-Science Inc 2010). The device is USB powered from a device (e.g. tablet or laptop) that makes the setup mobile and able to collect data in the field. To collect data, the device is clamped onto a leaf of a particular plant and the profile of the amount of light absorbed or reflected is captured as a spectrogram on the device for each position of the clamped leaf.

Several ambient factors influence the intensity and shape of the spectra, illumination being particularly important. For this reason, we collected data directly infields under similar lighting conditions.

We used the reflectance mode of operation of the spectrometer based on previous experiments where reflectance and absorption modes of operation gave the same performance for cassava leaves.

We collected data for plants aged 6 to 9 months from several cassava varieties including Nase 3, Nase 4, Nase 14, Nase 19, Alado Alado, Magana, Orera and NAROCass 2 (Nakabonge et al. 2018). For each variety, three plants were considered; and for each plant, three leaves were considered. For each leaf, two spectral readings were taken on each leaf lobe: one on the best part (least affected/non-symptomatic) and the worst part (most affected/symptomatic) (Fig. 5.1). Because the spectrometer takes readings on a small area of the plant about 2cm in diameter, readings for every leaf lobe were recorded in order to achieve a representative and reliable sampling. Note that this was considered during validation, such that we never trained and tested on data from the same plant. In total, 1,656 data points were collected for evenly distributed classes: healthy, CMD and CBSD.

Figure 5.1: Depiction of asymptomatic(good) and symptomatic(bad) part of a leaf

5.3. Experiments

(a) Visibly healthy (b) CBSD (c) CBSD (d) CMD (e) CMD

Figure 5.2: Example images of leaves of cassava manifesting the different diseases.
5.3. Experiments

Figure 5.1: Depiction of asymptomatic (good) and symptomatic (bad) part of a leaf.

Figure 5.2: Example images of leaves of cassava manifesting the different diseases.

Figure 5.3: Illustration for class-conditional means of Cassava spectral data not individual spectra. The left panel displays raw, full signal, the right panel shows the corresponding pre-processed spectra.

5.3.2 Data pre-processing

We follow the same pre-processing procedure defined in the previous chapter. In Figure 5.3, we illustrate the original and the final pre-processed spectrogram for the three classes. The resulting spectrogram correspond to 2,500 equally spaced feature dimensions which was quite high. Although dimensionality reduction could also be interpreted as optional pre-processing step, it is closely linked with training of the model, therefore we present them separately.

5.3.3 Training and validation

The data collection involved picking more than one sample from a particular plant, therefore it was important to choose a validation strategy that matched this condition in order to avoid training and testing on data from the same plant. We kept track of the class label (healthy, CBSD and CMD) as well as the unique plant labels (also called groups). During training, partitioning was based on plant groups and
the validation scheme was Shuffle-Group(s)-Out cross-validation.

We employed the standard Scikit-learn (Pedregosa et al. 2011) implementation of this cross-validation scheme for the algorithms that were implemented using Scikit-learn. In a similar way, this validation strategy was implemented for LVQ in MATLAB(R2016a) for the open source GMLVQ toolbox (Biehl 2017) that we employed for the GMLVQ algorithm. For all the models we train, we carry out a 10-fold cross-validation and average the performance over the folds. We employ parameter K=15 for the KNN algorithm, C = 1 for the linear SVC and 200 estimators for the Extra trees algorithm. For the GMLVQ algorithm we employ standard parameters used in the GMLVQ tool box which is available online (Biehl 2017).

5.4 Results

In this section we present results of training the LVQ family of methods and other standard algorithms (SVC and KNN) on the full spectral dataset and on the reduced dataset with different kinds of feature reduction: PCA, Chebyshev and peak methods. As a baseline we also applied a Convolutional Neural Network (CNN) using 1-D convolutional filters given the nature of the data. CNNs are models that have been shown to have superior performance on many tasks from computer vision, to natural language processing. One of the complexities of implementing CNNs is the choice of the architecture required for a particular problem.

In our case, for comparison with other base algorithms we employed a convolutional neural net (CNN) with an architecture based on the principles of popular models for image classification, but adapted to be suitable for 1D inputs. We use repeated convolution and ReLu blocks, with a max-pooling operation at the end of each block. A final fully connected softmax layer is used to perform the classification. This architecture is analogous to the VGG-16 architecture for 2D (image) inputs (Simonyan and Zisserman 2014), using the same principle that the initial layers are intended to capture ‘local’ patterns within the input, and the successive convolution and max-pooling blocks successively downsample the input so that more global patterns can also be captured. Note, however, that because in our case the model operates on 1D spectral data, there would be no way to utilise existing CNN models such as VGG-16 for 2D data directly as starting points for training; we therefore trained this model from scratch starting from a random initialisation.

5.4.1 Full spectral data

Our goal is two-fold: (1) to develop an algorithm that can perform well on spectral leaf data and (2) to engineer the algorithm with a reduced featureset to a comparable
performance. The first goal builds a baseline for building models for disease detection on cassava plants that are non-symptomatic and the second provides the base for design and implementation of low-cost devices that can use the narrow bands discovered as relevant in this study. The spectrometer we used costs in the order of thousands of dollars, and we aim to build one costing tens of dollars.

To investigate model performance with the full spectral data (goal 1) we pre-processed the data by truncating the signal at the extreme ends of the spectrogram and using the band 400 - 900 nm, as earlier described. We trained six algorithms, three from the family of LVQ, two from Scikit-learn and a CNN. The LVQ algorithms included Generalized Learning Vector Quantization (GLVQ) that does not train for relevances of the feature vector, Generalized Relevance Learning Vector Quantization (GRLVQ) that is similar to GLVQ but that trains a vector $\lambda$ which represents the relevance of the features and GMLVQ which uses a matrix of relevances as described in an earlier section. The other three algorithms were a Linear Support Vector Classifier (SVC), K-Nearest Neighbour and CNN.

Table 1 shows the classification accuracies for the six different algorithms with the full dataset. We obtained good performance for the GMLVQ, SVC and CNN algorithms, with the SVC algorithm showing the best performance on this dataset. One important consideration to note in Table 1 is that feature reduction techniques produce reduced featuresets that are not immediately amenable to calculating convolutions as is the case for CNNs, and performance thus degrades.

The profile of the wavelengths most relevant for the classification (Fig. 5.4) was derived from the diagonal of matrix $\Lambda$ from the GMLVQ algorithm. Results in Table 1 indicate that this provides some level of advantage over the SVC algorithm which performed best with the original dataset.

![Figure 5.4: Feature relevance as quantified by diagonal elements of $\Lambda$, cf. Eq. (5.3), for original spectra as feature vectors.](image-url)
Projections of the feature vectors onto the leading eigenvectors of $\Lambda$ allows depicting the spatial location of the training data points in relation to the prototypes per class see (Fig. 5.5). This 2-D representation shows good placement of the prototypes in the space of the training data.

Figure 5.5: Visualization of the dataset depicting the three major classes in the dataset plotted as projections of feature vectors (original spectra) on the two leading eigenvectors of GMLVQ relevance matrix.

5.4.2 Reduced feature space

For the reduced features we tried the three methods described in the previous sections: PCA, Chebyshev and a method based on truncating the peaks of the relevance profile produced from training the algorithms on the full spectral data.

The challenge with feature reduction is to reduce the features from the 2500 features to a suitable number $N$ that can represent the full spectrogram and still perform relatively well on the dataset. To determine a suitable $N$, we ran experiments of a set of different $N$s and plotted the accuracy of the GMLVQ algorithm and compared these with the Chebyshev method and PCA methods for different values of $N$ (Fig. 5.7).

The results of this experiment were an optimal $N$ for PCA of 30 coefficients and Chebyshev of close to 200 components. We then trained our set of five algorithms on these reduced featuresets of the data. Table 1 shows results of the PCA and Chebyshev methods of feature reduction.

The peak method of selecting a set of wavelengths from the relevance profile of the full spectral data is a fairly intuitive way of reducing the set of features. For
5. Matrix relevance learning for multi-class classification with spectral data

Projections of the feature vectors onto the leading eigenvectors of Λ allows depicting the spatial location of the training data points in relation to the prototypes per class see (Fig. 5.5). This 2-D representation shows good placement of the prototypes in the space of the training data.

Figure 5.5: Visualization of the dataset depicting the three major classes in the dataset plotted as projections of feature vectors (original spectra) on the two leading eigenvectors of GMLVQ relevance matrix.

5.4.2 Reduced feature space

For the reduced features we tried the three methods described in the previous sections: PCA, Chebyshev and a method based on truncating the peaks of the relevance profile produced from training the algorithms on the full spectral data.

The challenge with feature reduction is to reduce the features from the 2500 features to a suitable number \( N \) that can represent the full spectrogram and still perform relatively well on the dataset. To determine a suitable \( N \), we ran experiments of a set of different \( N \)s and plotted the accuracy of the GMLVQ algorithm and compared these with the Chebyshev method and PCA methods for different values of \( N \) (Fig. 5.7).

The results of this experiment were an optimal \( N \) for PCA of 30 coefficients and Chebyshev of close to 200 components. We then trained our set of five algorithms on these reduced featuresets of the data. Table 1 shows results of the PCA and Chebyshev methods of feature reduction.

The peak method of selecting a set of wavelengths from the relevance profile of the full spectral data is a fairly intuitive way of reducing the set of features. For this method we used a threshold on the relevance profile to select the 30 most relevant features from the profile to train our algorithms. Results of the peak method

Figure 5.6: Visualization of GMLVQ prototypes of the original spectra

Figure 5.7: Performance of classifiers based on N Principal Component (left) and n coefficients in the polynomial representation (right panel).

this method we used a threshold on the relevance profile to select the 30 most relevant features from the profile to train our algorithms. Results of the peak method
Table 5.1: Overall accuracy in original feature space (400 - 900 nm) and when applying dimensional reduction techniques.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Original space</th>
<th>PCA</th>
<th>Chebyshev</th>
<th>Peaks</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLVQ</td>
<td>0.605</td>
<td>0.998</td>
<td>0.719</td>
<td>0.702</td>
</tr>
<tr>
<td>GRLVQ</td>
<td>0.845</td>
<td>0.856</td>
<td>0.738</td>
<td>0.699</td>
</tr>
<tr>
<td>GMLVQ</td>
<td>0.949</td>
<td>1.000</td>
<td>0.996</td>
<td>0.952</td>
</tr>
<tr>
<td>Linear SVC</td>
<td>0.972</td>
<td>0.971</td>
<td>0.753</td>
<td>0.709</td>
</tr>
<tr>
<td>KNN</td>
<td>0.786</td>
<td>0.848</td>
<td>0.815</td>
<td>0.823</td>
</tr>
<tr>
<td>CNN</td>
<td>0.969</td>
<td>0.935</td>
<td>0.840</td>
<td>0.330</td>
</tr>
</tbody>
</table>

Figure 5.8: Selection of features with diagonal relevances (GMLVQ) above a threshold.

are shown in Fig. 5.8 and Table 1. We obtained the best performance with these reduced feature sets for the GMLVQ algorithm with the PCA feature reduction method performing the best overall for all five algorithms.

A key idea of our experimentation was to determine how well we could derive the relevance profile from the original spectral data profile using the reduced features. By Eq. 5.7, we reconstructed the relevance profile from the reduced features for PCA method using five coefficients (very few) and 30 coefficients (optimal performance) according to Fig. 5.9.

The results of this back transformation are shown in Fig. 5.9. The resultant shapes of the relevance profiles for all wavelengths tended to follow the shape of the relevance profile of the original spectral data with the bi-modal distribution of relevance. In a general way this justifies our choice of $N$ for the PCA method.
5.5 Discussion

We presented a method of diagnosing disease from plant leaves in the field using spectral data which is different from previous methods based solely on image data. This method provides the first step in our search for a method that can be used to diagnose disease in leaves before they are visibly symptomatic. In future work we intend to look at spectral data collected from a more controlled environment where plants can be inoculated and data collected before they are visibly symptomatic. For this work, however, we obtained a comparable level of classification accuracy in the difficult problem of classifying between a healthy cassava plant and those affected by CMD and CBSD. Although the use of spectrometry for classification is not a new idea, its novel application for this particular problem is highlighted.

The results showed improved classification accuracy when using a reduced featureset, particularly when PCA was used for feature reduction. The spectral data were very noisy and it is possible that the feature reduction removed most noise in the signal. As expected, performance of the CNN degrades with reduction in the number of features since deep neural networks particularly excel with a large amount of data and many features. We also observed an interesting change in top performance with SVC performing best for the full spectrum data, but the roles changed and GMLVQ performed best consistently with all the other reduced representations of the featureset. One explanation for this is that with the reduced featureset, GMLVQ is able to train a more reliable relevance profile, which in turn enhances performance of the algorithm. However, with SVC, which may not calibrate for relevance profiles of the features, the decision boundaries are thrown off with the relatively fewer data points.
Figure 5.10: Receiver operating characteristic curves for one class vs All (multi-class problem). Top-left panel shows CBSD vs All and CMD vs All in the original feature space (400 - 900nm). Top-right panel shows CBSD vs All and CMD vs All with reduced features (peak selection). The bottom panel shows Healthy vs All both in the original space and reduced features (peak selection). The solid lines refer to AUC in original feature space (400 - 900nm) while the dashed lines refer to AUC with peak selection between 500 - 600nm.

One aspect of our future work is to build a low-cost smartphone add-on spectrometer. This work provides a feature band of spectra that is most relevant for diagnosing these serious diseases. One possible way of building the spectrometer is by use of specific diodes (an additional cost of $5 – 8) that are sensitive at those particular relevant feature wavelengths and building a light emitting, absorption and measurement system around that.